Price Fluctuation in Online Leasing

Björn Feldkord(✉), Christine Markarian, and Friedhelm Meyer Auf der Heide

Computer Science Department, Heinz Nixdorf Institute, Paderborn University,
Fürstenallee 11, 33102 Paderborn, Germany
{bjoernf,chrissm,fmadh}@mail.uni-paderborn.de

Abstract. Current theoretical attempts towards understanding real-life leasing scenarios assume the following leasing model. Demands arrive with time and need to be served by leased resources. Different types of leases are available, each with a fixed duration and price, respecting economy of scale (longer leases cost less per unit time). An online algorithm is to serve each arriving demand while minimizing the total leasing costs and without knowing future demands. In this paper, we generalize this model into one in which lease prices fluctuate with time and are not known to the algorithm in advance. Hence, an online algorithm is to perform under the uncertainty of both demands *and* lease prices. We consider different adversarial models and provide online algorithms, evaluated using standard competitive analysis. For each of these models, we give deterministic matching upper and lower bounds.

Keywords: Online algorithms · Leasing · Infrastructure problems · Parking permit problem · Ski-rental problem

1 Introduction

Over the years, *leasing* has become a widely adopted business model in many markets. Companies needing access to expensive equipment have been avoiding the risk of *buying* resources, that may soon become obsolete, and *leasing* them for limited periods instead. As a result of its flexibility and various advantages, leasing has been used in many forms and employed in plenty of applications. Despite its prominence, the first *theoretic* study that aimed towards better understanding leasing scenarios has been introduced in 2005, by Meyerson [13].

Meyerson has proposed the first theoretic leasing model, phrased as a simple daily-life problem: the Parking Permit Problem, described as follows. Each day, depending on the weather, we have to either use the car (if it is rainy) or walk (if it is sunny). In the former case, we must have a valid parking permit, which we choose among K different types of permits (leases), each having a different duration and price. At any time, lease prices respect *economy of scale* such that a longer lease costs less per unit time. The goal is to buy a set of leases in

This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Centre "On-The-Fly Computing" (SFB 901).

X. Gao et al. (Eds.): COCOA 2017, Part II, LNCS 10628, pp. 17–31, 2017.
https://doi.org/10.1007/978-3-319-71147-8_2

order to cover all rainy days while minimizing the total cost of purchases and without using weather forecasts. This simple problem, in which a single resource (a permit) is leased, to cover arriving demands (rainy days), captures the main notion of online leasing. There have been a series of works that extend this notion to more sophisticated problems such as involving multiple resources [1,4,12,14] or more flexible demands (demands that need not be covered immediately) [11].

All these models assume that resources have *fixed* prices that do not change over time. Nevertheless, due to their dynamic nature, most markets are likely to face fluctuations in their resource prices. These may often be hard to predict and hence leasing decisions tend to be more critical and challenging.

Our Contribution. In this paper, in pursuit of better understanding these challenges, we incorporate the lack of this knowledge into the leasing model by allowing lease prices to change over time. These are given by an adversary and not known to the algorithm in advance. Hence, an online algorithm is to perform under the uncertainty of both demands (no weather forecast) *and* lease prices.

To evaluate our algorithms, we use the standard *competitive analysis* in which an online algorithm is compared to the optimal offline algorithm which is optimal and knows the entire sequence of demands and lease prices in advance. Given an input sequence σ, let $\mathcal{C}_A(\sigma)$ and $\mathcal{C}_{OPT}(\sigma)$ denote the cost incurred by an algorithm A and an optimal offline algorithm OPT, respectively. Algorithm A is c-competitive if there exists a constant α such that $\mathcal{C}_A(\sigma) \leq c \cdot \mathcal{C}_{OPT}(\sigma) + \alpha$ for all input sequences σ.

It is easy to see that, without any restrictions on the prices, an adversary can set the competitive ratio to an arbitrary large number. Thus, we weaken the power of the adversary by imposing restrictions on how prices change. We define the following adversarial models. The first adversary sets the prices for each lease type k within an interval $[C_k, f \cdot C_k]$ for some constant f (Sect. 3). The second adversary allows the price of a lease to only change by at most 1 between any two consecutive days (Sect. 4). We also consider these adversaries with the assumption that demands are given to the algorithm in advance (Sect. 5). For each of these models, we give deterministic matching upper and lower bounds. We further generalize some of these results to problems involving multiple resources (Sect. 6).

2 Related Work and Background

In this section, we give an overview of the related literature and provide some definitions needed throughout the rest of the paper.

Related Work. A standard assumption in most resource allocation problems has been the permanence of the resources purchased. Once a resource is bought, it is assumed it can be used any time in the future without inducing further costs that can be influenced by time or number of uses.

In pursuit of better economies of scale, a number of models have been introduced. These include the Buy-at-Bulk model [3] in which cost varies with the capacity a resource provides (larger capacity is cheaper per unit) and the

Rent-or-Buy model formulated as the Ski-Rental problem, defined as follows. Each day, a skier is to decide whether to buy or rent skis while minimizing total skiing costs and without knowing when the skiing season ends [8,9].

A generalization of the Ski-Rental problem is the Parking Permit Problem described earlier [13], such that the number of leases K is set to 2. Meyerson has given a deterministic $\mathcal{O}(K)$-competitive and a randomized $\mathcal{O}(\log K)$-competitive algorithm along with matching lower bounds. He has also introduced the leasing variant of the online Steiner Forest problem, known as Steiner Tree Leasing. The goal in the classical online Steiner Forest problem is to select a subset of edges of minimum weight such that each pair of arriving nodes is connected. Steiner Forest Leasing asks to lease edges for K different durations/prices such that an edge can be used only during its lease period (must lease it again should we need it at a later step) and the goal is to connect each arriving pair (called terminals) for the current step, while minimizing the total leasing costs. Meyerson has given a randomized $\mathcal{O}(\log n \log K)$-competitive algorithm for Steiner Forest Leasing, where n represents the number of nodes in the input graph and K the number of available leases. Recently, Bienkowski et al. [6] have proposed a deterministic algorithm with $\mathcal{O}(K \log s)$-competitive ratio for Steiner Tree Leasing (a special case of Steiner Forest Leasing in which there is a fixed root node to which arriving requests that are single nodes must be connected), where s denotes the number of terminals and K the number of available leases.

Inspired by Meyerson's work, Anthony and Gupta [4] have generalized his idea to other infrastructure problems: (metric) Facility Location, Set Cover, and Steiner Tree. An analogous definition to Steiner Forest Leasing is given to each of these infrastructure leasing problems, known as (metric) Facility Leasing, Set Cover Leasing, and Steiner Tree Leasing, respectively. Anthony and Gupta have showed an interesting connection between infrastructure leasing problems and stochastic optimization problems that leads to approximation algorithms for the offline variants of these problems. They have given an $\mathcal{O}(K)$ (where K is the number of available leases), $\mathcal{O}(\log n)$ (where n is the number of elements in the Set Cover instance), and $\mathcal{O}(\min(K, \log n))$ (where n is the number of nodes in the graph and K the number of available leases) approximation for these variants, respectively.

Nagarajan and Williamson [14] have later improved the $\mathcal{O}(K)$-approximation for (metric) Facility Leasing to an (offline) 3-approximation and have given an $\mathcal{O}(K \log n)$-competitive algorithm for its online variant, where n is the number of clients. Kling et al. [10] have extended the results by Nagarajan and Williamson [14] for the online variant by removing the dependency on n (and thereby on time). They have given an $\mathcal{O}(l_K \log(l_K))$-competitive algorithm, where l_K is the maximum lease length. Abshoff et al. [2] have given the first online algorithm for Set Cover Leasing and have improved previous results for online variants of Set Cover. Li et al. [11] have extended Meyerson's leasing model by introducing demands that need not be served upon arrival, but have deadlines. Hu et al. [7] have extended the Parking Permit Problem to a two-dimensional variant in which lease types have lengths and capacities.

A variant of the Ski-Rental Problem in which the ski-rental price changes over time has been introduced by Bienkowski [5]. He has studied several models differing in the knowledge given to the algorithm in terms of the duration of the skiing season and has given algorithms with competitive ratios up to constant or logarithmic factors optimal.

In this paper, we generalize the leasing framework given by the Parking Permit Problem and introduce pricing models differing in how lease prices change over time.

Background. We briefly introduce the formal definition of the original Parking Permit Problem and a variant we often use for our analysis.

Parking Permit Problem: We are given a set of K lease types, defined by prices $C_1, \dots, C_K$ and durations $l_1, \dots, l_K$. A day t is covered, if a lease of some type k is purchased on day t', such that $t' \le t \le t' + l_k - 1$. The goal is to cover all given rainy days with minimal costs.

Interval Model: In this variant, a lease type k always starts at times $i \cdot l_k + 1$ for $i \in \mathbb{N}_0$. We refer to an interval of the form $[i \cdot l_k + 1, (i+1) \cdot l_k]$ as an interval of type k. In addition, we also assume that all lease intervals align with each other (i.e., l_k is a multiple of l_{k-1}).

Throughout the paper, we often refer to the deterministic algorithm for the Parking Permit Problem by Meyerson [13] and so we restate it here. The algorithm assumes the Interval Model and reads as follows: 'As soon as the optimum offline algorithm (using only the schedule seen so far) would purchase a lease type k, the online algorithm buys it'. This algorithm has an $\mathcal{O}(K)$-competitive ratio (Theorem 3.1 in [13]).

In the original Parking Permit Problem, the Interval Model could be assumed with the loss of at most 4 in the competitive ratio (Theorem 2.2 in [13]). In our model, however, this is not true in general due to the changes in lease prices. Hence, we argue about the loss whenever we make this assumption.

In our model, we use C_k to refer to the lowest price which occurs for a lease type k on a given sequence. The restrictions on the occurring price changes are described at the beginning of each respective section.

3 Arbitrary Prices

We consider the following problem. Each day, an adversary determines whether it is rainy or sunny. It also provides the algorithm with the prices of the leases for the current day. Prices of leases are allowed to change essentially in an arbitrary way between two consecutive days. The only restriction is that prices for each lease type k are within an interval $[C_k, f \cdot C_k]$ for some constant f. We give deterministic matching lower and upper bounds for this problem. These bounds depend on the parameters f and K. We also show that the dependency on f can be avoided when the adversary is replaced by a simple stochastic process.

For the lower bound below, we adopt ideas from the lower bound for the original Parking Permit Problem while incorporating the maximum price change.

The main idea is to only give a low price for a lease on the first day of its duration, such that an online algorithm can not yet make the decision to buy it, if it is a longer and hence expensive lease.

Theorem 1 *(Lower Bound). Every deterministic algorithm for the Parking Permit Problem with arbitrary prices has a competitive ratio of at least $\Omega(f \cdot K)$.*

Proof. Let ALG be an online algorithm for the Parking Permit Problem with arbitrary prices. We assume the interval model and define our K lease types as follows. For the durations we set $l_1 = 1$ and $l_k = 2Kf^3 \cdot l_{k-1}$ for $k > 1$. The costs are set to $C_k = (2f^2)^k$. This implies that $2f^2 \cdot C_k = C_{k+1}$ for all $k < K$. We construct an input sequence such that a rainy day occurs every time the current day is not covered by ALG. On the first day of each interval of type k, the price is C_k. For the other days in such an interval, the price is $f \cdot C_k$.

For every k, we define x_k as the number of times the online algorithm buys a lease type k on the first day of the corresponding interval. In the same way we define n_k for the intervals of type k where the online algorithm buys the corresponding lease on the second day or later (this is not possible for $k = 1$, hence $n_1 = 0$). From this we directly get $\mathcal{C}_{Alg} = \sum_k (x_k \cdot C_k + n_k \cdot fC_k)$ for the costs of the online algorithm. Now let y_k be the number of intervals of type k containing at least one rainy day and where ALG does not buy a lease type k on the first day. A possible solution is to cover each interval of type k with a non-zero number of rainy days by a lease type k. In the case that ALG buys a lease type k on the first day of such an interval, there will not be more rainy days in this interval, hence the optimal solution can cover it with a lease type 1. Therefore we have for all k: $\mathcal{C}_{Opt} \leq x_k \cdot C_1 + y_k \cdot C_k$.

We define r_k as the number of type k intervals with a non-zero number of rainy days for which ALG does not buy a lease of any type $j \geq k$ (hence $r_1 = 0$). We can show that the algorithm has to pay at least KfC_k for each of these intervals. For $k = 2$, if ALG does not buy a lease type 2 or higher it needs $^{l_2}/_{l_1}$ leases of type 1 to cover the interval. Therefore the costs for this interval are at least

$$^{l_2}/_{l_1} \cdot C_1 \geq 2K \cdot f^3 \cdot C_1 \geq KfC_2.$$

In the same way, consider an interval of type $k > 2$ and denote by $\mathcal{C}_{Alg}(k-1)$ the costs ALG pays to cover an interval of type $k-1$. By induction we know that $\mathcal{C}_{Alg}(k-1) \geq KfC_{k-1} \geq C_{k-1}$ if it does not cover the whole interval with a lease type $k-1$. Therefore the costs for covering the type k interval are at least

$$^{l_k}/_{l_{k-1}} \cdot C_{k-1} \geq 2K \cdot f^3 \cdot C_{k-1} \geq KfC_k.$$

Using the above estimations and $y_k = r_k + \sum_{j \geq k} n_j + \sum_{j > k} x_j$ we get

$$\mathcal{C}_{Opt} \leq x_k \cdot C_1 + y_k \cdot C_k \text{ for all } k$$

$$\Rightarrow (K-1)f \cdot \mathcal{C}_{Opt} \leq \sum_{k=2}^{K} (x_k f C_1 + y_k f C_k)$$

$$
\begin{aligned}
&\le \sum_{k=2}^{K} x_k C_k + \sum_{k=2}^{K} \left(r_k f C_k + \sum_{j \ge k} n_j f C_k + \sum_{j=k+1}^{K} x_j f C_k \right) \\
&\le 2 \cdot \mathcal{C}_{Alg} + \sum_{k=1}^{K} \left(n_k f \sum_{j \le k} C_j + x_k \sum_{j \le k} C_j \right) \\
&\le 4 \cdot \mathcal{C}_{Alg}. \qquad \square
\end{aligned}
$$

Note that this bound also holds without the assumption of the interval model, as we show next. The optimal solution can treat the problem as in the proof, only buying leases on the first day of a given interval. The algorithm produces a solution which is not necessarily aligned with the intervals. However, leases which are bought for a low price are already aligned with those intervals, while leases which are bought for the high price can be replaced by two leases for at most the same price. Hence, any solution of an algorithm against the given sequence in the non-interval model can be transformed into a solution of at most twice the costs for the interval model.

It should also be noted that the sequence of prices is only increasing for a fixed lease type within an interval and always repeats itself. From the proof, we can observe that even knowing this sequence of prices in advance does not improve the possible performance of any online algorithm in this setting.

Next we show how to achieve a matching $\mathcal{O}(f \cdot K)$ upper bound for the problem. To this end, we show that any c-competitive algorithm for the original Parking Permit Problem can be transformed into a $(c \cdot f)$-competitive algorithm for the Parking Permit Problem with arbitrary prices.

Theorem 2 *(Transformation). Let* Alg *be any c-competitive algorithm for the Parking Permit Problem.* Alg *can be transformed into a $(c \cdot f)$-competitive algorithm* Alg' *for the Parking Permit Problem with arbitrary prices.*

Proof. Let Alg be any c-competitive algorithm for the Parking Permit Problem and let I be any instance of the Parking Permit Problem with arbitrary prices. We construct Alg' as follows. For each lease type i in I we fix its prices to the first price for lease type i revealed by the adversary. Then we run Alg while purchasing online the leases it outputs. Let Opt be the cost of the optimal solution for A with arbitrary prices. The cost of our solution constructed is upper bounded by $c \cdot Opt'$, where Opt' is the cost of the optimal solution based on the first prices, fixed by the algorithm. Clearly, we have that $Opt' \le f \cdot Opt$ and so the theorem follows. $\square$

It turns out that when prices are restricted to be only non-increasing with time, it is possible to have a competitive ratio independent of f. The following theorem shows that the deterministic algorithm by Meyerson achieves that.

Theorem 3 *(Upper Bound). For the Parking Permit Problem with arbitrary, non-increasing prices, the deterministic algorithm in [13] is $\mathcal{O}(K)$-competitive.*

Proof. We assume the interval model and lose a factor 2 in the competitiveness, as follows. Consider the optimum solution for the original problem. Any lease type k in the optimum solution intersects and thus can be covered by at most two consecutive leases of the same type in the interval model. The first of these leases is bought on the same day the optimum lease is bought and hence the online algorithm pays exactly what the optimum algorithm does. Since prices are non-increasing, the cost of the second lease is at most the cost of the optimum lease.

We use induction over the lease types. For $k = 1$, either the online algorithm pays 0 or the price for the interval which the optimum also has to pay. For $k > 1$, we observe that the induction hypothesis directly implies $\mathcal{C}_{Alg} \leq (k-1)\mathcal{C}_{Opt}$ if the optimum does not buy a permit of type k for this interval. Otherwise, we have $\mathcal{C}_{Alg} \leq (k-1)\mathcal{C}_{Opt}$ by induction until the day such that the optimum would have decided to buy a permit of type k. But then the algorithm at most pays the same price for this permit as the optimum, since the price can only be non-increasing. It follows that for every interval of type k that $\mathcal{C}_{Alg} \leq k \cdot \mathcal{C}_{Opt}$ if this interval was the whole input which implies the competitive ratio. □

So far we have seen that when price curves are given by an adversary, the maximum price change within an interval reflects directly on the competitive ratio of any online algorithm, even if the price curves have a simple repeating structure and are known in advance.

However, if the specific curve from the lower bound is replaced by curves in which good prices for our algorithm appear more often, rather than just forcing a decision on a specific day in the sequence, then we may get a competitive ratio independent of f.

We demonstrate this effect by introducing a simple variant of the problem in which the price continues to drastically change, but the times at which it changes are determined by a stochastic process.

For a lease type $i > 1$, the prices C_i and $f \cdot C_i$ are available. The price C_i is chosen with probability $p > 0$ and $f \cdot C_i$ is chosen with probability $(1-p)$. The price of the first lease type is assumed to be a constant C_1. In this way, the resulting prices can still form the same pattern as in the deterministic lower bound.

The goal here is to provide an algorithm with competitive ratio independent of the maximum price change f. In order to achieve this, we propose an algorithm that tries to avoid buying a lease at a high price and compensates with an expected waiting time $\Theta(\frac{1}{p})$ instead. Our algorithm assumes the interval model and is described as follows.

Algorithm. Let k be the lease type with maximum C_k such that $C_k \leq \frac{1}{p}C_1$. As long as no lease type $i > k$ would be bought by the optimal solution, we cover all requests with leases of type k which we buy as soon as the low price is available. We use leases of type 1 to cover the time of waiting for this price. As soon as the optimal solution would have bought a lease type $i > k$, we buy it on the next time step where the price is low and as before cover all requests in the waiting period with leases of type 1.

We show in the following theorem that the algorithm above achieves a competitive ratio independent of the maximum price change f for the stochastic price model.

Theorem 4 *(Upper Bound)***.** *There exists an $\mathcal{O}\left(K + \frac{1}{p}\right)$-competitive algorithm for the stochastic price model.*

Proof. Let k be the maximum lease type with $C_k \leq \frac{1}{p}C_1$. Replacing every lease types 1 up to k in the optimal solution with a lease type k has expected costs of at most

$$\sum_{t=1}^{\infty}(1-p)^{t-1}p((t-1)C_1 + C_k) = \frac{1-p}{p}C_1 + C_k \leq 2\frac{1}{p}C_1.$$

Now consider the behavior of the algorithm on lease types i with $C_i > \frac{1}{p}C_1$. The algorithm only attempts to buy such a lease if the optimal algorithm has bought it as well. The expected costs of the algorithm are at most

$$\sum_{t=1}^{\infty}(1-p)^{t-1}p((t-1)C_1 + C_i) = \frac{1-p}{p}C_1 + C_i \leq 2C_i.$$

By induction, it follows that the costs of the algorithm for these lease types are at most $2K \cdot \mathcal{C}_{Opt}$.

It is easy to see that this analysis also holds for the non-interval model, since the adversary is assumed to always pay the low price and hence the costs of a solution in the interval model with this assumption are at most 2 times the costs of the optimal solution in the non-interval model. □

The competitive ratio above tends to infinity if p becomes very small. However, if p becomes too small, a ratio $(p + (1-p)f)K$ can always be achieved by applying the algorithm by Meyerson. This ratio is also superior in case p is close to 1. More precisely, the stated ratio of $K + \frac{1}{p}$ is smaller if $p \in [\frac{1}{2}(1 - \sqrt{\frac{K(f-1)-4}{K(f-1)}}), \frac{1}{2}(1 + \sqrt{\frac{K(f-1)-4}{K(f-1)}})]$.

4 The Progressive Model

The results in the previous section raise the question of whether the problem is hard to solve in general or these results can be improved when more restrictions are imposed on the adversary. Hence, we consider the following problem. Each day, an adversary determines whether it is rainy or sunny. It also provides the algorithm with the prices of the leases for the current day. Unlike in the previous section, prices can now change by at most 1 between two consecutive days.

The resulting prices are much closer to an almost continuous behavior which occurs on several digital goods, especially those in which prices are not determined by a single seller but emerge from a high frequency trade as in the stock market.

In what follows, we give deterministic matching lower and upper bounds. In this model, C_k refers to the lowest price occurring for a lease of type k.

Theorem 5 *(Lower Bound)***.** *Every online algorithm for the Parking Permit Problem with progressive prices has a competitive ratio of at least $\Omega(K + \frac{l_K}{C_K})$.*

Proof. The lower bound K follows directly from that of the Parking Permit Problem. As for l_K/C_K, consider an instance of the problem with two leases ($K = 2$). Let $C_1 = l_1 = 1$. The price for the first lease type remains fixed, while the price C_2 for the second lease increases until time $\frac{l_2}{2}$ and then decreases again. There will be no rainy days during the first $\frac{l_2}{4}$ steps. We choose $C_2 < \frac{l_2}{4}$. For the online algorithm, consider the following possibilities:

1. The online algorithm does buy the lease type 2 during the first $\frac{l_2}{4}$ steps. Then there will be no further rainy days, and the optimal solution pays 0. Hence, the competitive ratio is unbounded.
2. The online algorithm does not buy the lease type 2 during the first $\frac{l_2}{4}$ steps. Then there will be $\frac{l_2}{2}$ rainy days starting from the $\frac{l_2}{4} + 1$st step and the costs are at least $C_2 + \frac{l_2}{4}$ for the online algorithm. The optimal costs are C_2.

This sequence can be repeated infinitely often and even works without a model with fixed intervals since a lease type 2 can never cover 2 complete blocks of rainy days. □

Note that the sequence of prices is again independent of the algorithm's behavior and repeats itself, implying that it does not help the algorithm if the prices are known in advance. Despite the restriction on the prices, notice that the maximum price change within the duration of a lease still reflects in the competitive ratio of any algorithm as illustrated in the lower bound above.

Theorem 6 *(Upper Bound)***.** *For the Parking Permit Problem with progressive prices, the deterministic algorithm by Meyerson is $\mathcal{O}(K + \frac{l_K}{C_K})$-competitive.*

Proof. We make two assumptions: (1) the interval model and (2) $2l_{k-1} \leq l_k$ which also implies $2C_{k-1} \leq C_k$. We show next that these assumptions lead to a loss of at most a factor 4 in the competitiveness. Note that the assumption for (1) holds only because we compare our online algorithm to the optimal offline algorithm which assumes no price changes. Clearly, the cost of this optimal offline algorithm is a lower bound for the cost of the actual optimal offline algorithm for the problem.

For (1), assume the optimal solution buys a lease type k. If we fix the intervals in which this lease can be bought, we may replace it by at most two leases of the same length. The costs of the optimal solution increase by a factor at most 2 since we assume that prices do not change for the optimal solution.

As for (2), we eliminate some of the lease types from the original problem as follows. We visit the leases one by one in decreasing length order. We keep the lease with the highest length and start eliminating the leases that do not satisfy $2l_{k-1} \leq l_k$. Now, consider a lease l_i in the optimal solution, which was eliminated. We replace l_i by the next highest lease l_j which is not eliminated. Due to economy of scale, we have that $\frac{C_j}{l_j} \leq \frac{C_i}{l_i}$ and since $l_j \leq 2l_i$, we get $C_j \leq 2C_i$ and hence lose a factor at most 2 in the competitive ratio.

Now, we use induction over the lease types to show that the algorithm pays at most $(k+2\frac{l_K}{C_K})\cdot c_k$ for a lease type k, where C_k is what the optimal algorithm pays at least to buy such an interval.

For $k=1$, the online algorithm pays the same as the optimum. For $k>1$, the algorithm pays at most $(k-1+2\frac{l_{k-1}}{C_{k-1}})\cdot C_{k-1}$ by induction hypothesis. For k, the algorithm pays $(k-1+2\frac{l_{k-1}}{C_{k-1}})\cdot C_{k-1}$ by induction until the day the optimum would have decided to buy a lease type k, on which the algorithm pays at most C_k+l_k. Hence the algorithm pays a total of at most $(k-1+2\frac{l_{k-1}}{C_{k-1}})\cdot C_{k-1}+C_k+l_k$. By substituting $2C_{k-1}\leq C_k$ and $2l_{k-1}\leq l_k$, we get $(k+2\frac{l_k}{C_k}))\cdot C_k$. □

5 Full Weather Forecast

The algorithms presented thus far are faced with the uncertainty of both future demands *and* price changes. To better understand the effect of price fluctuation, we impose further restrictions on the adversary.

For any two lease types i and j, we define $r_{ij}(t)$ to be the ratio of price of i to price of j on day t (lease price ratio). The adversary can change lease prices such that for any two days these ratios remain unchanged. Moreover, the algorithm is aware of all demands in advance (i.e., has access to *full* weather forecast).

Note that we already determined that giving the online algorithm full knowledge of the prices while rainy days arrive online does not change the competitive ratio of the problem since our lower bounds always use a fixed price curve seen so far.

In what follows, we give deterministic lower and upper bounds for arbitrary and progressive prices.

Theorem 7 *(Lower Bound)***.** *Every deterministic algorithm for the Parking Permit Problem with full weather forecast and arbitrary prices has a competitive ratio of at least* $\Omega(f)$*.*

Proof. We consider an instance with 2 leases. We set $l_1=C_1=1$, $l_2=3f$ and $C_2=2f$. The requests occur at the f last time steps. We adapt the price sequence according to the following cases:

1. The online algorithm buys the second lease at the first day of the sequence. Its costs are therefore $2f$. We drop prices by a factor f and the optimal solution pays at most 2.
2. The online algorithm does not buy the second lease at the first day. We increase prices by a factor f. The optimal solution can buy a lease type 2 on the first day.

Since we always change the two prices by the same factor at the same time, the lease price ratio stays the same throughout the sequence. □

Theorem 8 *(Lower Bound)***.** *Every deterministic algorithm for the Parking Permit Problem with full weather forecast and progressive prices has a competitive ratio of at least* $\Omega(\frac{l_K}{C_K})$*.*

Proof. We consider an instance with 2 leases. A sequence of length l_2 is divided into four phases as illustrated in Fig. 1. Rainy days occur exactly on all days of the third phase. The price curve is chosen between 2 versions based on the behavior of the online algorithm in the first phase.

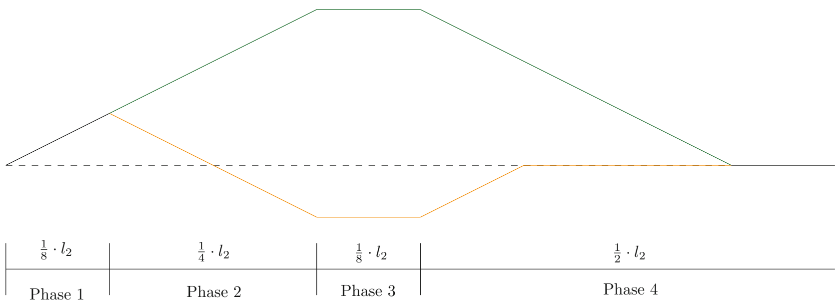

Fig. 1. Illustration of the price curve during one period of the lease type 2.

The online algorithm either buys the lease type 2 before or after price has risen above $\frac{1}{16} \cdot l_2$ in the first phase. If the online algorithm bought it before that, we choose the lower (orange) curve for the prices and enforce a difference of at least $\frac{1}{8} \cdot l_2$. Otherwise we enforce the same difference in prices by choosing the upper (green) curve. Therefore the costs of both algorithms differ by $\Theta(l_2)$.

We ensure that covering the rainy days with leases of the first type is never a superior option by setting $l_1 = 1$ and ensuring $C_2 < \frac{1}{8} l_2 C_1$. The price curve of the lower lease behaves such that the lease price ratio stays the same throughout the sequence. □

Theorem 9 *(Upper Bound)***.** *For the Parking Permit Problem with full weather forecast, there is a deterministic algorithm with competitive ratio $\mathcal{O}(f)$ for arbitrary prices and $\mathcal{O}(1 + \frac{l_K}{C_1})$ for progressive prices.*

Proof. The algorithm assumes that the price of each lease type k is the first price seen for this type and constructs an optimal offline solution OPT_E based on these prices. It then buys online the leases in OPT_E. For the analysis, we set these prices to their minimum and this is possible since for any two lease types i and j, $r_{ij}(t)$'s remains unchanged for all days and so an optimal offline solution comprises of the same leases for any two days, given the same schedule of rainy days. Let $\mathcal{C}_{Opt_E}$ be the cost of OPT_E after setting the prices of leases to their minimum. We assume the interval model and hence lose a factor at most 2 by the same argument as in the proof of Theorem 6. Moreover, we analyze the algorithm over the first l_K time steps. Let $\mathcal{C}_{Alg}$ and $\mathcal{C}_{Opt}$ denote the cost of the online algorithm and the cost of the optimum algorithm, based on the original lease prices, respectively. Clearly, $\mathcal{C}_{Opt_E} \leq \mathcal{C}_{Opt}$. For arbitrary prices, the competitive ratio follows from $\mathcal{C}_{Alg} \leq f \cdot \mathcal{C}_{Opt_E}$. For progressive prices, the algorithm pays l_K more for each lease bought in OPT_E. Suppose OPT_E contains $|OPT_E|$ leases. Then, $\mathcal{C}_{Opt_E} = \sum_{s=1}^{|OPT_E|} C_{j(s)}$, where $j(s)$ denotes the type $1, ..., K$ of the corresponding lease. The competitive ratio then follows from

$$\mathcal{C}_{Alg} \leq \sum_{s=1}^{|OPT_E|} (C_{j(s)} + l_K) \leq (1 + \frac{l_K}{C_1}) \cdot \mathcal{C}_{Opt_E}.$$

□

6 Generalizations

The results so far address price fluctuation of a *single* resource (a permit). It is natural to ask whether these results can be generalized to multiple resources. Hence, we dedicate this section to infrastructure leasing problems with resource prices changing over time. Resource prices are determined, as before, by the arbitrary, progressive, and full weather forecast (arbitrary/progressive) models.

Corollary 1 *(Transformation).* *Let A be any infrastructure leasing problem with any c-competitive algorithm* Alg*.* Alg *can be transformed into a $(c \cdot f)$-competitive algorithm and a $(c \cdot (1 + \frac{l_K}{C_K}))$-competitive algorithm for A with arbitrary and progressive prices, respectively.*

Proof. The same arguments as those in Theorems 2 and 6 for a single resource hold for any infrastructure leasing problem with multiple resources. □

Corollary 2 *(Transformation).* *Let A be any infrastructure leasing problem with any (offline) c-approximation algorithm* Alg *when demands are known in advance.* Alg *can be transformed into a $(c \cdot f)$-competitive algorithm for A with full weather forecast and arbitrary prices and a $(c + \frac{l_K}{C_1})$-competitive algorithm for A with full weather forecast and progressive prices.*

Proof. The same argument as that in Theorem 9 for a single resource holds for any infrastructure leasing problem with multiple resources. □

Notice that the competitive ratio for the progressive model in Corollary 1 is l_K/C_K *times* the ratio attained by an algorithm for the original problem. In the Parking Permit Problem, however, we showed that it is possible to have an *additive* factor of l_K/C_K instead ($\mathcal{O}(K + \frac{l_K}{C_K})$). We observe that while the results for the other adversarial models can easily be generalized to *any* infrastructure leasing problem and *any* corresponding algorithm, generalizing the results for the progressive model seems to require a closer look at the characteristics of the specific algorithm/problem at hand. As an example, we examine the deterministic algorithm for the Facility Leasing problem by Nagarajan and Williamson [14].

In Facility Leasing, we are given a set of m potential facility locations F and a set of n potential clients U in a metric space. On each day t, the adversary gives a set $D_t \subset U$ of clients that must be connected to a facility which is in lease on day t. There are K different possible types for leasing a facility and the cost of leasing a facility $f \in F$ with lease type i is c_i^f. Connecting a client to a facility incurs a cost equal to the distance between the two. The goal is to connect each arriving client while minimizing the total leasing costs and connecting costs.

Nagarajan and Williamson [14] proposed an $\mathcal{O}(K \cdot \log n)$-competitive algorithm for Facility Leasing, based on the primal-dual scheme. We modify their algorithm to achieve an $\mathcal{O}((K + \frac{l_K}{C_K}) \log n)$-competitive ratio for Facility Leasing with progressive prices, as follows.

On the first day we fix the prices of all leases/facilities to their corresponding prices given by the adversary for that day and run the primal-dual algorithm by Nagarajan and Williamson based on these prices. Then we purchase online the leases/facilities the primal-dual algorithm outputs. Clearly, we pay for each of the purchased leases the corresponding price for the day we buy. While most of the analysis does carry over, it suffices to just modify Lemma 5.4 in [14]. The proof of Lemma 5.4 can be modified according to the following observation. The cost of opening facilities is measured such that every dual variable pays into K leases at the same time, one for each type. For every lease type k bought, the actual price might be up to l_k higher than the one accounted for in the dual solution. Hence, by using similar arguments as in the proof of Theorem 6, we conclude the following.

Corollary 3 *(Upper Bound)***.** *There is an $\mathcal{O}((K + \frac{l_K}{C_K}) \log n)$-competitive algorithm for Facility Leasing with progressive prices.*

We conjecture that this technique can be applied to algorithms that work similar to the primal-dual algorithm in [14]. In particular, an important characteristic is that the algorithm does not spend more on smaller leases than on a longer lease within the lease period of that lease. This characteristic seems to appear in all of the deterministic algorithms for the problem with competitive ratio dependent on K. This is the result of covering an interval of type K with all lease types having costs equal to that of the longest lease.

7 Concluding Remarks and Future Work

In this paper, we initiate the study of price fluctuation in online leasing. Our results imply that the effect of price changes is always apparent, even when demands are known in advance and the ratio between the prices remains fixed over time. The table below shows a comparison between the bounds attained for the two pricing models and the knowledge required by the online algorithm beforehand.

As a summary of our results, we may conclude that the maximum price change does reflect in both pricing models, but only as an additive term in the progressive model.

For both models, full knowledge about the occurring prices does not improve the competitive ratio. However, knowledge of the rainy days (demands) does remove the dependency on the number of lease types if, in addition, the ratio

	Arbitrary	Progressive
Unknown rainy days/ Unknown prices	$\Theta(f \cdot K)$	$\Theta(K + \frac{l_K}{C_K})$
Unknown rainy days/ Known prices	$\Theta(f \cdot K)$	$\Theta(K + \frac{l_K}{C_K})$
Known rainy days/ Unknown prices	$\Theta(f)$	$\mathcal{O}(1 + \frac{l_K}{C_1}), \Omega(\frac{l_K}{C_K})$

between the prices remains fixed. Nevertheless, the dependency on the maximum price change remains.

From the previous section we conclude that the bounds for the Parking Permit Problem reflect in other leasing problems as well, as we showed either through general transformations or by example of specific algorithms. We also conjecture that the lower bounds carry over in a similar fashion.

At this point, one may want to look at some other pricing models, arising, for instance, from specific actual markets or other stochastic processes. Competitive ratios independent of the maximum price change may then be possible. Moreover, the latter does not seem to be possible even by extending the current randomized approaches for leasing problems and thus developing new randomization techniques could be an interesting next step.

References

1. Abshoff, S., Kling, P., Markarian, C., Meyer auf der Heide, F., Pietrzyk, P.: Towards the price of leasing online. J. Comb. Optim., 1–20 (2015)
2. Abshoff, S., Markarian, C., Meyer auf der Heide, F.: Randomized online algorithms for set cover leasing problems. In: Zhang, Z., Wu, L., Xu, W., Du, D.-Z. (eds.) COCOA 2014. LNCS, vol. 8881, pp. 25–34. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-12691-3_3
3. Andrews, M., Zhang, L.: Wavelength assignment in optical networks with fixed fiber capacity. In: Díaz, J., Karhumäki, J., Lepistö, A., Sannella, D. (eds.) ICALP 2004. LNCS, vol. 3142, pp. 134–145. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27836-8_14
4. Anthony, B.M., Gupta, A.: Infrastructure leasing problems. In: Fischetti, M., Williamson, D.P. (eds.) IPCO 2007. LNCS, vol. 4513, pp. 424–438. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72792-7_32
5. Bienkowski, M.: Price fluctuations: to buy or to rent. In: Bampis, E., Jansen, K. (eds.) WAOA 2009. LNCS, vol. 5893, pp. 25–36. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12450-1_3
6. Bienkowski, M., Kraska, A., Schmidt, P.: A deterministic algorithm for online steiner tree leasing. Algorithms and Data Structures. LNCS, vol. 10389, pp. 169–180. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62127-2_15
7. Hu, X., Ludwig, A., Richa, A., Schmid, S.: Competitive strategies for online cloud resource allocation with discounts: the 2-dimensional parking permit problem. In: 35th IEEE International Conference on Distributed Computing Systems, ICDCS 2015, pp. 93–102. Columbus, OH, USA, 29 June–2 July 2015
8. Karlin, A.R., Manasse, M.S., Rudolph, L., Sleator, D.D.: Competitive snoopy caching. Algorithmica **3**, 77–119 (1988)

9. Karp, R.M.: Online algorithms versus offline algorithms: how much is it worth to know the future? In: Algorithms, Software, Architecture - Information Processing 1992, Proceedings of the IFIP 12th World Computer Congress, vol. 1, pp. 416–429. Madrid, Spain, 7–11 September 1992
10. Kling, P., Meyer auf der Heide, F., Pietrzyk, P.: An algorithm for online facility leasing. In: Even, G., Halldórsson, M.M. (eds.) SIROCCO 2012. LNCS, vol. 7355, pp. 61–72. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31104-8_6
11. Li, S., Mäcker, A., Markarian, C., Meyer auf der Heide, F., Riechers, S.: Towards flexible demands in online leasing problems. In: Xu, D., Du, D., Du, D. (eds.) COCOON 2015. LNCS, vol. 9198, pp. 277–288. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21398-9_22
12. Markarian, C., Meyer auf der Heide, F.: Online resource leasing. In: Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, pp. 343–344. Donostia-San Sebastián, Spain, 21–23 July 2015
13. Meyerson, P.: The parking permit problem. In: Proceedings of 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005), pp. 274–284. Pittsburgh, PA, USA, 23–25 October 2005
14. Nagarajan, C., Williamson, D.P.: Offline and online facility leasing. Discrete Optim. **10**(4), 361–370 (2013)

http://www.springer.com/978-3-319-71146-1

Combinatorial Optimization and Applications
11th International Conference, COCOA 2017, Shanghai, China, December 16-18, 2017, Proceedings, Part II
Gao, X.; Du, H.; Han, M. (Eds.)
2017, XVIII, 529 p. 88 illus., Softcover
ISBN: 978-3-319-71146-1